A Search Approach Based on Query Similarity in Content-Centric Networks

Jiun-Yu Tu*, Chih-Lin Hu†, and Han Hu†
*Dept. of Computer Science and Information Engineering,
Southern Taiwan University of Science and Technology, Tainan 71005, Taiwan, R.O.C.
†Dept. of Communication Engineering, National Central University, Taoyuan 32001, Taiwan, R.O.C.
duketu@stust.edu.tw; clhu@ieee.org; huhan@g.ncu.edu.tw

***Abstract*—Based on the Content-Centric Network (CCN) architecture, a data content can be stored at routing nodes along the transmission path on which the data content transmitted from the content source repository back to the requester. Therefore, while the routing nodes receive the same content requests again, the routing nodes that have stored the queried contents can answer the requests directly by responding with the requested contents from their own storage area. Thereby it eliminates the cost of sending the content requests to the content source repositories again, and thus improves the overall network transmission efficiency. In the CCN network, a routing node refers to the Forwarding Information Base (FIB) of its own forwarding engine – the FIB function is similar to the routing table used in the traditional IP network, for determining where to send out the request packets. This study proposes a Search Approach based on Query Similarity (SAQS), which analyses the information of the temporary data stored in the routing node itself, calculates the weight of the temporarily stored contents for coming query requests, and selects the node that is most likely to store temporarily the queried contents. Finally, through simulation, the proposed SAQS is compared with other query forwarding methods. Validating by different simulation cases, our proposed SAQS exhibits better content search hit rate, while the number of hops and delay time can still remain in satisfactory results.**

***Index Terms*—naming content, naming prefix, content search, content-centric network, CCN.**

I. Introduction

The traditional IP network transports packets across Internet by an end-to-end communication model. When sending packets, a querying host retrieves the IP addresses of content repositories which are the destinations of sent querying packets, and then a content requester sends querying packets directly to corresponding content repositories. The querying packets flow across the Internet under delivery of network routing nodes along paths toward each corresponding target repository. The routing nodes in traditional IP network would not keep in mind relations between each individual querying packet. On the other hand, when content repositories response with required contents, content packets flow back to corresponding querying hosts. The routing nodes locating along backward paths transport content packets without being aware of contents carried upon data packets. The routing nodes would not keep anything for next similar or same queries.

The CCN network [1] has content query and exchange based on content rather than the IP address where the location of the data in the traditional IP network. When transporting queried contents, routing nodes may temporarily store some copies of the queried contents. So, when the same query requests come, the routing nodes do not have to send out the requests to the content repositories. Instead, the routing nodes can responded directly back the stored queries to the corresponding end requesters. This design can reduce the responding time that the end requesters waiting for retrieving their interested contents, and can reduce the system burden of the entire network and content repositories.

This study proposes a search approach based on query similarity that analyzes the temporarily stored contents in its own Content Store (CS) to determine which neighboring node is the best next hop to send the coming content queries. Through the simulation, it is shown that the proposed content queries forwarding mechanism has better performance than the shortest-path routing (SPR) strategy in terms of cache hit ratio, average hop distance, and average delay time.

The rest part of this study is organized as follows: the brief background of CCN and the previous literature are mentioned in the section II; the proposed mechanism is described in the Section III; the simulation environment and results are listed in the Section IV; finally, a conclusion is written in Section V.

II. Background and Related Works

A. Background

In CCN networks, the naming of content is similar to the format of Uniform Resource Locator (URL), that is, the naming of contents adapts a hierarchical manner, and each keyword is preceded by a slash ("/") symbol. For example, a content is named "/parc.com/picture/carA.jpg". When receiving a content query request with above naming string, a routing node in the CCN network would navigate inside the node itself according to the named-prefix of the naming string to determine if the routing node itself is occasionally storing the queried content. With the temporarily stored contents in some routing nodes, a query request has a higher possibility to be answered on the halfway toward the specific repositories.

There are two types of packets in a CCN network, which are the interest packets and the data packets. The interest packets have their purpose to carry a content query request while the data packets are used when corresponding contents are responded by the specific content repositories or by a routing node which is locating halfway toward the specific content repositories.

A CCN routing node is composed of three functional elements:

1 Content store (CS) - for temporarily storing contents. When a routing node receives a queried content data packet that is backward to a querying user, the routing node can decide to store temporarily the carried content in the CS.
2 Pending interest table (PIT) - for the interest packets which have not yet been completed. When a content repository returns the corresponding queried content data packet, a CCN routing node would transmit the data packet back to the user on the same path according to the record in the PIT.
3 Forwarding information base (FIB) - similar to the IP routing table using in routers in the traditional IP network. However, the information stored in an FIB in a CCN routing node is not the IP addresses, but the information of a next hop connected to the CCN routing node itself. FIB also plays the role of determining the route of interest packets.

B. Related Works

Among previous studies, [2], [3] mentioned that not all the name of outgoing interfaces of a FIB may match the naming of a querying request. When a routing node decides to discard an interest packet because of mismatching between the name of a query request and of outgoing interfaces merely, the performance of the whole CCN network would be degraded. Especially when a routing node adopts a single-port query forwarding policy. So, in [2], the authors decided to added the querying name carried in an interest packet to the current interface. When the querying name of the interest packet is added, the interest packet can be forwarded to a farther side to improve the probability of hunting the queried content. In [3], the authors proposed a method that a content repository node in a CCN network can either spread contents in a proactive manner or deliver contents passively. However, the manner of pro-actively spreading content information introduces significantly large traffic volume which would congests the network.

On the other hand, [4], [5] adopt multi-interfaces forwarding to improve the network efficiency. The lecture [4] proposed OMP-IF to forward the interest packets to multiple non-intersecting paths to reduce the network load while the lecture [5] discussed issues of single-interface transferring and multi-interfaces transferring. In the study, the conclusion is that adopting the shortest path routing (SPR) strategy [7] may cause the useless of queried contents that have been stored at the halfway CCN routing nodes. In addition, the nearest copies of queried contents maybe ignored by the shortest path strategy and generates more hunting costs.

III. Search for similarity query methods

This study focuses on improving the performance of forwarding decisions that the FIB of a routing node makes. Referring to the peer ranking mechanism demonstrated in the previous lecture [6], this study proposed an effective query forwarding strategy. By our proposed mechanism, the FIB of a CCN routing node keeps the naming information of each outgoing interfaces as well as hitting-rate vectors of the stored queried content that each neighbor nodes stored. According to the proposed hitting-rate vectors, the FIB of a routing node estimates the best outgoing interface to transfer interest packets to their next hops.

A. Named String Classification

A CCN network adopts a hierarchical data structure naming system. A named string of a CCN content data is composed of multiple name components, each name component is split by a slash symbol "/". For example, the composition of /parc.com/videos/WidgetA.mpg/v3/S0.

For getting assigning weightings to each keywords, the proposed mechanism assigns each levels of the named string into a class C_j. So, the named string above has classifications between each keywords as C_1(parc.com)/C_2(videos) /C_3(WidgetA.mpg)/C_4(v3)/C_5(s0). In the above instance, we are trying to find out the most possible faces to retrieve the required content chunk.

B. Query Weightings

By the proposed classification mechanism mentioned in above Section III-A, the FIB of a CCN routing node assigns weightings to each keywords from the leftmost word consequently in a degrading manner. Using above string as an example, assuming that the leftmost C_1(parc.com) has an assigned weighting, said 5, and the next one, C_2(videos), has an assigned weighting, said 4, and so on. As a result, the named string has its weighting vector, G=[5, 4, 3, 2, 1]. Considering the leftmost keyword 'parc.com' denotes that the content repository is residing under the network domain 'parc.com'. When hunting a queried content, the FIB has to find out the content repository first, and then find out matching contents from the content repository, thus, the leftmost keyword that denotes a network domain retrieves the highest weighting. The same concept is applied to each other keywords that degrades the assigned weightings consequently.

C. Cache Hitting-ratio Vector of Temporarily Stored Queried Contents

In this study, the proposed mechanism introduces a data structure, hitting-ratio vector H_i^*, into the FIB function component of a routing node in addition to the interface names listed in the FIB. The notation H_i^* represents the cache hitting-ratio of the temporarily stored contents in CS of each neighbouring routing nodes. By the named string classification mentioned above in Section III-A, each routing node enumerate total class and total number of temporary content in a CS to determine a cache hitting-ratio vector of a node. The notation y_j^i denotes the total number of categories C_j in the node i, then, z_j^i which is an element of a hitting-ratio vector of node i to each category C_j is derived by equation (1). Note that the hitting vector is represented as $H_i^* = [z_1^i, z_2^i, \cdots, z_L^i]$, where L denotes the total number of naming string categories stored in CS of the node i.

$$z_j^i = \frac{y_j^i}{\sum_{j=1}^{L} y_j^i} \tag{1}$$

D. Review Ratio Mechanism

In this study, an FIB forwarding engine utilizes the inner product of the query weighting matrix, G, and the hitting-ratio vector, H_i^*, to determine the best output face to forward a specific interest packet, as represented in equation (2).

$$H^* G = \Phi \tag{2}$$

Assume that there are M faces connecting to the routing node, S, which currently receives an interest packet. The routing node, S, establishes a matrix according to the contents stored in its own CS and the corresponding hitting-ratio vectors. Each column in a matrix is a hitting-ratio vector of one of the output faces for each data naming classification. Then, the FIB forwarding engine performs inner product operation to the query weighting matrix G to learn the best face to transport the coming interest packet. The result Φ vector of the output faces is derived in equation (3).

$$\begin{bmatrix} z_1^1 & z_2^1 & \dots & z_L^1 \\ z_1^2 & z_2^2 & \dots & z_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ z_1^M & z_2^M & \dots & z_L^M \end{bmatrix} \begin{bmatrix} G_1 \\ G_2 \\ \vdots \\ G_l \end{bmatrix} = \begin{bmatrix} \Phi_1 \\ \Phi_2 \\ \vdots \\ \Phi_M \end{bmatrix} \tag{3}$$

Finally, the FIB forwarding engine selects the best face which is the element obtaining the highest in Φ as the forwarding face.

IV. Experiment and analysis

A. Simulation Environment

This study invokes ccnSim network simulator to validate the proposed design. Table IV-A lists parameters introduced into performance simulation.

TABLE I
Parameters introduced into the simulation

Parameters	Description
Network topology	Random
Number of nodes	100
Number of Requesters	88
Number of data repositories	11
Transmission delay	1 ms
Total number of data content	10000 - 100000
Storage space	10 - 10000
Query forwarding strategy	SPR [7], SAQS
Data storage strategy	LEC [1], LCD [8], Prob [9]
Data replacement strategy	FIFO

B. Performance evaluating indexes

To evaluate performance of the proposed design, this study invokes three performance evaluating indexes, whose definitions are described as follows:

- Cache hit ratio: The cache hit ratio, η denotes an average rate that stored contents match interest packets at every routing nodes in a CCN network. Because a routing node in a CCN network checks its own CS first each time it receives an interest packet. When a stored content chunk fits the query, the counting of hitting rate would be increased by one. Otherwise, the counting of miss rate would be increased by one. Then, the Cache hit ratio is expressed as:

$$\eta = \frac{number of cache hits}{number of outgoing request} = \frac{hits}{miss + hits} \tag{4}$$

- Average Hop distance: The average hop distance, δ denotes the average hops after which interest packets can retrieve queried content chunk. The average hop distance is expressed as:

$$\delta = \frac{\sum_{i=0}^{k} h_i}{k}, \tag{5}$$

where k is the total number of interest packets generated by content requesters, and h_i denotes the hops that the $i-th$ interest packet went through.

- Average waiting time: The average waiting time, τ, denotes the average response delay while requesters send out their queries.

$$\tau = \frac{\sum_{i=0}^{k} t_i}{k}, \tag{6}$$

where t_i denotes total time escaped that every queries retrieve their corresponding querying contents, and k denotes the total number of interest packets.

C. Impact of temporary storage space

By the simulation, this study examines the performance improvement of our proposed SAQS mechanism related to the SPR strategy. The simulation introduced 10,000 content, and the temporary storage space is 10-10000.

As shown in Figure 1, 2, and 3, the η exposes a positive correlated relationship with the content storage space. That is, when a routing node increases its content storage space, η of the routing node would increase as well. Besides, both the average hop distance, δ, and the average waiting time, τ, exhibit a negative correlated relationships with the content storage space. That is, when a routing node increases its content storage space, both δ and τ decrease. The observation above illustrates that when a routing node increases its content storage space, more content could be stored at the node, thus, there is a higher probability that interest packets could be satisfied at the routing node, which causes a higher η. In addition, once interest packets have higher probability be satisfied at each individual routing node, the travel distance of each interest packet would be shortened. The shorter travel distance causes lower δ and τ.

In Figure 1, our proposed SAQS achieves a higher η than that the SPR does regardless of which kind of content storage and content replacement strategies. The proposed SAQS effectively leverages neighboring nodes to assist a routing node to hunt queried content quickly. In the simulation cases, our proposed SAQS introduces eight neighbors to each individual routing node in average. In other words, using the proposed SAQS mechanism, a routing node is surrounded by a plenty candidates of querying content. Further more, the proposed SAQS adopts multi-face query forwarding strategy. A routing node forwards interest packets to more than one neighbors. It increases the hitting probability of interest packets. While the SPR strategy adopts shortest path forwarding method, that is, a routing node forwards an interest packet to a next hop which is closest to the content source repository. It causes the fact that each routing node routes interest packets following fix paths. There is no possibility that an SPR routing node find out new shorter path to hunt a queried content.

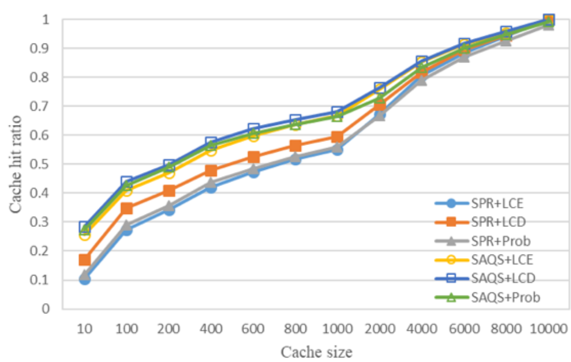

Fig. 1. Cache hit ratio when the amount of contents in the entire network: 10000 with first-in-first-out replacement strategy

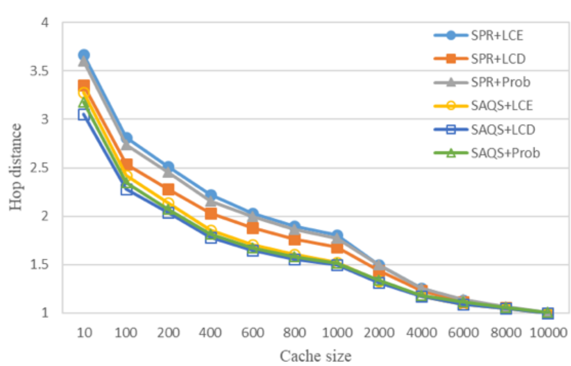

Fig. 2. Average hop distance when the amount of contents in the entire network: 10000 with first-in-first-out replacement strategy

V. Conclusion

This paper proposes a request forwarding mechanism which is based on query similarity (SAQS) under a CCN network. The primer concept of our proposed design is to analyse the information stored in the CSs of neighbouring nodes related to an individual routing node, and then, an individual

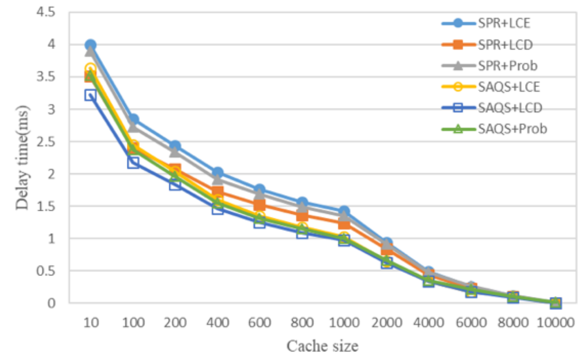

Fig. 3. Average delay time when the amount of contents in the entire network: 10000 with first-in-first-out replacement strategy

routing node refers to the analysis result to forward queries to the corresponding faces that most likely possess the specific queried content. The performance of our proposed SAQS is mainly compared with SPR strategy which adopts shortest path method to find out the closest next hop an interest packet should go to. Under different simulation cases, our proposed SAQS has higher cached content hitting ratio, shorter average hop distance, and lower average waiting time than that the SPR strategy does.

Acknowledgment

*This work was supported in part by the Ministry of Science and Technology, Taiwan, under Contracts NSC100-2221-E-008-085-MY3 and MOST-105-2221-E-008-029-MY3.

References

[1] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, Dec. 2009, pp. 1–12.
[2] H. Dai, J. Lu, Y. Wang, and B. Liu, "A two-layer intra-domain routing scheme for named data networking," in 2012 IEEE Global Communications Conference, Dec. 2012, pp. 2815–2820.
[3] A. Udugama, X. Zhang, K. Kuladinithi, and C. Goerg, "An on-demand multi-path interest forwarding strategy for content retrievals in ccn," in 2014 IEEE Network Operations and Management Symposium, May. 2014, pp. 1–6.
[4] G. Rossini and D. Rossi, "Evaluating ccn multi-path interest forwarding strategies," Computer Communications, pp. 771–778, Apr. 2013.
[5] C. L. Hu, Y. H. Chang, and K. F. Huang, "Searching with feature similarity in unstructured peer-to-peer networks," in 7th International Conference on Ubi-Media Computing, pp. 65–71, Jul. 2014.
[6] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems amp; Workshops, Mar. 2008, pp. 60:1–60:10.
[7] R. Nakamura and H. Ohsaki, "Performance Comparison of Shortest-Path Routing and Optimal Detour Routing in Content-Centric Networking," in 2016 IEEE 40th Annual Computer Software and Applications Conference, Jun. 2016.
[8] G. Zhang, Y. Li, and T. Lin, "Caching in information centric networking: A survey," Computer Networks, pp. 3128–3141, Nov. 2013.
[9] I. Psaras, W. K. Chai, and G. Pavlou, "Probabilistic in-network caching for information-centric networks," in Proceedings of the Second Edition of the ICN Workshop on Information-centric Networking, Aug. 2012, pp. 55–60.